UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/531,397	03/21/2000	Joseph C. Ballantyne	3797.81466	6866

28319 7590 08/13/2004

BANNER & WITCOFF LTD.,
ATTORNEYS FOR MICROSOFT
1001 G STREET , N.W.
ELEVENTH STREET
WASHINGTON, DC 20001-4597

EXAMINER	
ALI, SYED J	

ART UNIT	PAPER NUMBER
2127	

DATE MAILED: 08/13/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 09/531,397	**Applicant(s)** BALLANTYNE, JOSEPH C.
	Examiner Syed J Ali	**Art Unit** 2127

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _13 May 2004_.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _1-8,13-19,21-25 and 27-30_ is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) _1-8,13-19,21-25 and 27-30_ is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) □ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application (PTO-152)

6) □ Other: _____.

DETAILED ACTION

1. This office action is in response to the amendment filed May 13, 2004. Claims 1-8, 13-19, 21-25, and 27-30 are presented for examination.

2. The text of those sections of Title 35, U.S. code not included in this office action can be found in a prior office action.

Claim Rejections - 35 USC § 103

3. **Claims 1-7, 16-19, 21, 23-25, and 27-30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gulick (USPN 6,421,702) in view of Nakamura (USPN 5,437,047) in view of Reiffin (USPN 6,330,583).**

4. As per claim 1, Gulick teaches the invention as claimed, including a method of scheduling CPU resources comprising the steps of:

using a counter to determine when to allocate the CPU resources (col. 5 lines 14-21); and

instructing the CPU to allocate resources in real-time (col. 4 lines 49-67).

5. Nakamura teaches the invention as claimed, including the following limitations not shown by Gulick:

allocating CPU resources via non-maskable interrupts issued from the counter (col. 3 lines 7-37).

6. Reiffin teaches the invention as claimed, including the following limitations not shown by Gulick or Nakamura:

instructing an interrupt controller to allocate the CPU resources (col. 4 lines 1-13).

7. It would have been obvious to one of ordinary skill in the art to combine Gulick with Nakamura since Gulick fails to specify the type of interrupt that is issued by the timer. Although Gulick does teach the use of non-maskable interrupts, these interrupts are issued from the operating system such that an executing task may be preempted by a higher priority task (Abstract; col. 11 lines 3-23). The function of the non-maskable interrupt issued by the operating system is similar to the interrupt issued by the timer, i.e., to terminate or preempt the executing task in favor of another task or a higher priority task. Thus, it would be beneficial to specify that the interrupt issued by the timer is also non-maskable, such that it is guaranteed that the interrupt service routine is executed. Thus, the combination of Gulick and Nakamura provides issuing a non-maskable interrupt from a timer (or counter) to preempt an executing task. Additionally, it would have been obvious to one of ordinary skill in the art to add Reiffin to the combination of Gulick and Nakamura since both Gulick and Nakamura teach issuing an interrupt from a timer (or counter) directly to the interrupt input of the CPU. This limits the specific functionality that could potentially be implemented within the interrupt controller. Specifically, Gulick teaches an operating system issuing a non-maskable interrupt to the CPU in order to preempt a lower priority task with a higher priority task. The reason for this is that the determination as to which task is of higher priority must be made within some sort of software module. Since the interrupt issued within Gulick is strictly a hardware interrupt, such a determination cannot be made satisfactorily. Thus, to use an interrupt controller, as disclosed by Reiffin, would allow the interrupt service routine to be implemented within the interrupt controller. Such a modification would be a marked

improvement over the modified Gulick, since all interrupts could be issued from the timer.

8. As per claim 2, Gulick teaches the invention as claimed, including the method of claim 1 wherein only a portion of the CPU resources are allocated (col. 7 lines 3-29).

9. As per claim 3, Gulick teaches the invention as claimed, including the method of claim 1 wherein all of the CPU resources are allocated (col. 7 lines 3-29).

10. As per claim 4, Gulick teaches the invention as claimed, including the method of claim 2 wherein the CPU resources are allocated to at least one thread, and the CPU resources are allocated by determining a duration of time (col. 6 lines 8-17) and a periodicity for execution of said at least one thread (col. 5 lines 14-21).

11. As per claim 5, Gulick teaches the invention as claimed, including the method of claim 3 wherein the CPU resources are allocated to at least one thread, and the CPU resources are allocated by determining a duration of time (col. 6 lines 8-17) and a periodicity for execution of said at least one thread (col. 5 lines 14-21).

12. As per claim 6, Gulick teaches the invention as claimed, including the method of claim 1 wherein the counter is a performance counter (col. 5 lines 14-43).

13. As per claim 7, Gulick teaches the invention as claimed, including the method of claim 6 wherein the performance counter counts machine cycles in order to determine when to allocate the CPU resources (col. 5 lines 14-43).

14. As per claim 16, Gulick teaches the invention as claimed, including a method of scheduling resources on at least one microprocessor that includes at least one performance counter, and at least one CPU, said method comprising the steps of:

allowing the CPU to execute a first thread (col. 9 line 44 - col. 10 line 25);

using the performance counter to determine when to allocate the resources to a second thread on a real-time basis (col. 5 lines 14-21; col. 4 lines 49-67);

issuing a first interrupt from the from the performance counter to the CPU when it is time to allocate the resource to the second thread (col. 5 lines 14-21);

instructing the CPU to switch execution from the first thread to the second thread (col. 9 line 44 - col. 10 line 25);

instructing the CPU to stop execution of the first thread (col. 9 line 44 - col. 10 line 25); and

allocating resources to the second thread (col. 9 line 44 - col. 10 line 25).

15. Nakamura teaches the invention as claimed, including the following limitations not shown by Gulick:

the interrupts are non-maskable (col. 3 lines 7-37);

the CPU storing first current state information regarding execution of the first thread (col. 3 lines 7-37); and the CPU restoring second current state information regarding execution of the second thread (col. 3 lines 38-46).

16. Reiffin teaches the invention as claimed, including the following limitations not shown by either Gulick or Nakamura:

the microprocessor includes at least one programmable interrupt controller, wherein the performance counter issues interrupts to the programmable interrupt controller when it is time to allocate CPU resources to the second thread (col. 4 lines 1-13); and

instructing the programmable interrupt controller to issue the second interrupt to the CPU that instructs the CPU to switch execution from the first thread to the second thread (col. 4 lines 1-13).

17. As per claim 17, Reiffin teaches the invention as claimed, including the method of claim 16 wherein the programmable interrupt controller is an APIC. The discussion of claim 16 discusses the programmable interrupt controller, and an APIC is defined to be a programmable interrupt controller that can handle interrupts from and for multiple CPUs. Therefore, since Reiffin teaches that the interrupt controller is for use in a multiprocessor system (Abstract), Reiffin inherently teaches that the programmable interrupt controller is an APIC.

18. As per claim 18, Gulick teaches the invention as claimed, including the method of claim 17 wherein the microprocessor is selected from the group consisting of: a Pentium 4GB, a Pentium Pro 64GB, a Pentium MMX 4GB MMX, a Pentium II 4GB MMX, a

Pentium II 4GB MMX KNI, a Celeron 4GB MMX, a Xeon PII 64GB MMX and a Xeon PIII 64GB MMX KNI (col. 4 lines 19-32).

19. As per claim 19, Gulick teaches the invention as claimed, including a computer-readable medium having computer-executable instructions stored for performing steps comprising:

using a scheduler to control execution of at least one thread based on an interrupt (col. 9 line 44 - col. 10 line 25); and

using at least one counter to issue a first interrupt to the CPU to notify the scheduler when to switch execution of said at least one thread on a real-time basis (col. 5 lines 14-21; col. 4 lines 49-67).

20. Nakamura teaches the invention as claimed, including the following limitations not shown by Gulick:

the interrupts are non-maskable (col. 3 lines 7-37).

21. Reiffin teaches the invention as claimed, including the following limitations not shown by either Gulick or Nakamura:

the interrupts are issued by an interrupt controller, wherein the interrupt controller relays the interrupt to the scheduler to switch execution of threads (col. 4 lines 1-13).

22. As per claim 21, Gulick teaches the invention as claimed, including the computer-readable medium of claim 19 wherein said least one counter is a performance counter and counts CPU cycles (col. 5 lines 14-43).

23. As per claim 23, the modified Gulick does not specifically disclose the computer-readable medium of claim 20 further comprising instructions for executing said at least one thread at a highest IRQ level. However, "Official Notice" is taken that it would have been obvious to one of ordinary skill in the art to place the interrupting thread at a highest IRQ level since that would ensure that the interrupting task is serviced without being preempted by another task. To that end, threads of a highest priority may be guaranteed Quality of Service, such that tasks of a highest priority are serviced accordingly.

24. As per claim 24, Reiffin teaches the invention as claimed, including the computer-readable medium of claim 20 further comprising instructions for executing said at least one thread in a transparent manner so that at least one operating-system process is unaware of the execution of said at least one thread (col. 4 lines 41-59).

25. As per claim 25, Gulick teaches the invention as claimed, including the computer-readable medium of claim 24 further comprising instructions for executing all of said operating-system process and all of said at least one threads as a single real-time thread (col. 1 line 48 - col. 2 line 35).

26. As per claim 27, Gulick teaches the invention as claimed, including the computer-readable medium of claim 19 further comprising instructions for allocating at least a portion of a CPU's resources to an operating-system process and using the remaining CPU resources for execution of said at least one thread (col. 7 lines 4-29).

27. As per claim 28, Gulick teaches the invention as claimed, including the computer-readable medium of claim 27 further comprising instructions for releasing the CPU resources back to the operating-system process when said at least one thread finishes execution (col. 7 lines 18-29).

28. As per claim 29, Gulick teaches the invention as claimed, including the computer-readable medium of claim 27 further comprising instructions for releasing the CPU resources to another thread when said at least one thread finishes execution (col. 7 lines 18-29).

29. As per claim 30, Gulick teaches the invention as claimed, including the computer-readable medium of claim 19 further comprising instructions for allocating a predetermined number of CPU cycles for execution of an operating-system process and using the remaining CPU cycles for execution of said at least one thread (col. 7 lines 4-29).

30. **Claims 8 and 22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gulick in view of Nakamura in view of Reiffin in view of Patterson et al. (previously cited) (hereinafter Patterson).**

31. As per claim 8, Patterson teaches the invention as claimed, including the following limitations not shown by Gulick, Nakamura, or Reiffin:

the method of claim 6 wherein the performance counter counts executed computer instructions (col. 2 line 66 - 3 line 43).

32. It would have been obvious to one of ordinary skill in the art to combine Gulick, Nakamura, and Reiffin with Patterson since within the framework of a real time application environment, restricting the interrupt controller to be triggered by only certain events prevents the environment from being modified to suit particular needs. For example, a real time task may be required to perform a certain amount of work before it is preempted, in addition to having a deadline. To that end, by preempting the task at the expiration of the period, rather than allowing the task to complete a minimum amount of work may cause performance of later tasks to suffer. Allowing the counter to be incremented based on any type of regularly occurring event, such as a computer instruction, allows the capabilities of the system to be modified to suit any number of specific needs.

33. As per claim 22, Patterson teaches the invention as claimed, including the computer-readable medium of claim 19 wherein said at least one counter is part of a CPU and counts executed computer instructions (col. 2 line 66 - 3 line 43).

34. **Claims 13-15 are rejected under 35 U.S.C. 103(a) as being unpatentable over Gulick in view of Nakamura.**

35. As per claim 13, Gulick teaches the invention as claimed, including a method of scheduling resources on at least one microprocessor that includes a CPU and a device, the method comprising the steps of:

using the device to determine, in response to a first interrupt, when to allocate the resources in real-time (col. 5 lines 37-43);

causing the device to issue a second interrupt to the CPU when it is time to allocate the resources (col. 5 lines 14-21); and

causing the CPU to allocate the resources in response to the second interrupt (col. 5 lines 14-21).

36. Nakamura teaches the invention as claimed, including the following limitations not shown by Gulick:

the interrupts are non-maskable (col. 3 lines 7-37).

37. It would have been obvious to one of ordinary skill in the art to combine Gulick with Nakamura since Gulick fails to specify the type of interrupt that is issued by the timer. Although Gulick does teach the use of non-maskable interrupts, these interrupts are issued from the operating system such that an executing task may be preempted by a higher priority task (Abstract; col. 11 lines 3-23). The function of the non-maskable interrupt issued by the operating system is similar to the interrupt issued by the timer, i.e., to terminate or preempt the executing task in favor of another task or a higher priority task. Thus, it would be beneficial to specify that the interrupt issued by the timer is also non-maskable, such that it is guaranteed that the interrupt service routine is executed. Thus, the combination of Gulick and Nakamura provides issuing a non-maskable interrupt from a timer (or counter) to preempt an executing task.

38. As per claim 14, Gulick teaches the invention as claimed, including the method of claim 13 wherein the device is a performance counter (col. 5 lines 14-43).

39. As per claim 15, Gulick teaches the invention as claimed, including the method of claim 13 wherein the device is a timer (col. 5 lines 14-21).

Response to Arguments

40. Applicant's arguments filed May 13, 2004 have been fully considered but they are not persuasive.

41. Applicant argues on page 8, "*In contrast to the invention recited in Claim 1, Gulick discloses real-time interrupts that cause an operating system to provide execution time slices for pending isochronous...tasks.*" Applicant adds, "*Gulick discloses use of non-maskable interrupts only for terminating isochronous tasks that fail to finish executing within their specified maximum duration. [Col. 11, lines 3-23]. Such termination of isochronous tasks is different than preemption of tasks by higher priority tasks.*"

42. Examiner respectfully disagrees. Primarily, Gulick does support the preemption of a lower priority task by a higher priority task (col. 9 lines 15-33). Gulick schedules lower priority tasks as "standard slices" that are preemptable, while higher priority tasks are assigned "quick slices" that are non-preemptable. While the interrupt service routine used in preemption does not utilize a non-maskable interrupt, Gulick does teach the use

of non-maskable interrupts for termination of a task. When Gulick is considered in combination with Nakamura, support can be found for issuing a non-maskable interrupt for preemption of a task of higher priority. Specifically, Gulick teaches periodically issuing an interrupt to check if a higher priority task has arrived, and if so, preempting the lower priority task such that the higher priority task may run. Although this interrupt is not explicitly set forth as being non-maskable, Nakamura demonstrates why it would be beneficial to have the interrupt be non-maskable, as discussed below (paragraph 44) in connection with the motivation to combine Gulick and Nakamura. Additionally, Applicant's arguments concerning whether the "*termination of isochronous tasks is different than preemption of tasks by higher priority tasks*", it should be noted that the claims are not related to interruption such that a higher priority task may be scheduled.

43. Applicant argues on page 9, "*Gulick's silence as to the type of interrupt by the timer would not motivate one of ordinary skill to substitute the non-maskable interrupts from Nakamura into the system disclosed by Gulick.*" Applicant presents additional arguments on page 10 related to the motivation to combine Gulick and Nakamura, concerning "*why a person of ordinary skill in the art would have been motivated to modify the system of Gulick as proposed in the Office Action.*"

44. While extensive motivation to combine Gulick and Nakamura has been provided above in paragraph 7, there exists additional intrinsic motivation to combine these references as well. Gulick teaches that if a lower priority task is preempted, a flag may be set such that the next time it is invoked, it may be designated as non-preemptable (col. 3 lines 3-10). When preemption is disabled in this manner, program contexts for tasks

that are running are unavailable to the kernel, such that if the task is abnormally terminated or otherwise suspended, the context may be lost. This is the problem sought to be addressed by Nakamura (col. 1 lines 31-45). Nakamura thus issues a non-maskable interrupt at a predetermined time interval to gather the program's context in case it is preempted, suspended, terminated, or otherwise ceases execution (col. 1 line 63 - col. 2 line 2; col. 2 lines 10-30). Thus, the use of a periodic non-maskable interrupt used in the preemption scheme of Gulick would allow the real-time processing to occur, while also protecting the executing task's context.

45. Applicant argues on page 9, "*Gulick...discloses using non-maskable interrupts for terminating execution of isochronous tasks that fail to execute within their specified maximum duration. But Gulick does not disclose using non-maskable interrupts issued from the operating system to preempt an executing task by a higher priority task.*" Applicant adds on page 10, "*Nakamura discloses using non-maskable interrupts for gathering and storing program-execution information from processors in a multi-processor environment. Accordingly, Gulick and Nakamura, either alone or in combination, fail to teach or suggest issuing a non-maskable interrupt from a timer [or counter] to preempt an executing task for the purpose of scheduling resources in real time.*"

46. These arguments are similar to those presented above in paragraphs 41 and 43. As discussion has been provided concerning how Gulick, Nakamura, and Reiffin meet the claim limitations, as well as how motivation exists for combining these references, the rejection stands as presented above. Additionally, Applicant's arguments related to

Gulick not using non-maskable interrupts to preempt an executing task by a higher priority task, the issue of priority is not presented in the claims, and is thus not of particular relevance to the argument.

Conclusion

47. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Syed J Ali whose telephone number is (703) 305-8106. The examiner can normally be reached on Mon-Fri 8-5:30, 2nd Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai T An can be reached on (703) 305-9678. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Syed Ali
August 5, 2004

MENG-AI T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100